

Learning When to Think in Latent Visual Space: Reasoning Mode Adaptive Reinforcement Learning

Jade Chein^{1*}, Jinpeng Ou^{1†}, Leo Wei^{1†}, Qingpei Guo¹, Jianing Li^{1‡}

¹Ant Group

3195794910@qq.com

{oujinpeng.ojp, qingpei.gqp, tiansuo.ljn}@antgroup.com

Abstract

Latent Visual Reasoning (LVR) unfolds reasoning in a continuous visual embedding space, markedly improving Vision-Language Models (VLMs) on vision-intensive tasks. However, our experiments reveal a hidden cost: LVR inevitably undermines VLMs’ strong textual reasoning ability, causing substantial degradation on logic-intensive tasks. This motivates us to unify both reasoning modes within a single VLM that adaptively selects the appropriate one. We propose **MoLaT** (Mixture of Latent and Textual Reasoning), a novel method that trains VLMs via reinforcement learning (RL) to select LVR and text-only reasoning adaptively. MoLaT is trained in two stages. Supervised fine-tuning endows the VLM with a dual-mode capability using mixed CoT data. Then, **MoLaT** introduces two complementary designs: 1) The **mode-adaptive advantage** decouples the mode-selection signal from per-rollout correctness reward at the token level: mode-selection tokens receive dedicated cross-mode gradients, while content tokens retain the group-relative advantage. 2) The **entropy-modulated advantage weight** uses sequence-level entropy as a difficulty proxy, modulating signal strength to prevent premature mode commitment. Experiments confirm that MoLaT improves both vision-intensive and logic-intensive abilities, validating adaptive mode selection as key to general-purpose reasoning VLMs.

1 Introduction

Multimodal large language models (MLLMs) have made remarkable progress in jointly processing visual and textual information (Li et al., 2024; Bai et al., 2025b; Wang et al., 2024b). Early efforts construct visual chain-of-thought (Visual CoT) data to guide multi-step reasoning in the textual space (Shao et al., 2024a; Xu et al., 2024).

Subsequent work advocates *thinking with images*, actively cropping, annotating, or editing intermediate images and re-encoding them to inject visual context into generation (Hu et al., 2024; Su et al., 2025; Zheng et al., 2025; Hong et al., 2025). While effective for logic-intensive reasoning and general question answering, these approaches transfer poorly to **vision-intensive tasks** such as maze solving, fine-grained visual search, and spatial relation understanding, whose bottleneck lies in visual perception rather than linguistic logic (Wu and Xie, 2024; Tong et al., 2024).

To overcome this limitation, **Latent Visual Reasoning (LVR)** (Li et al., 2025b) trains VLMs to generate latent states approximating query-relevant visual tokens, allowing reasoning to unfold directly in the visual embedding space. Subsequent work advances LVR along multiple axes, including curating interleaved vision-text reasoning data (Zhang et al., 2025b,a), introducing novel architectural components (Dong et al., 2025; Zhang et al., 2025b), and designing dedicated training objectives (Wu et al., 2026; Wang et al., 2025b), collectively yielding substantial gains on vision-intensive tasks.

However, existing LVR methods treat latent visual reasoning as universally applicable, overlooking that the optimal reasoning mode differs across problems. **Not all problems benefit equally from latent visual reasoning.** Our preliminary experiments (Section 2) show that LVR incurs catastrophic drops on logic-intensive tasks; ablation confirms that the degradation exceeds typical SFT forgetting, indicating active interference with pre-existing language-reasoning pathways (Figure 1). A concurrent work, SwimBird (Tong et al., 2026), attempts to guide mode selection through multi-mode SFT data; however, its selection behavior depends on training-data distributional priors shaped by human biases, and the model never receives explicit feedback on mode merits. This raises a key

*This work is done during internship at Ant Group.

†Equal contribution.

‡Corresponding Author

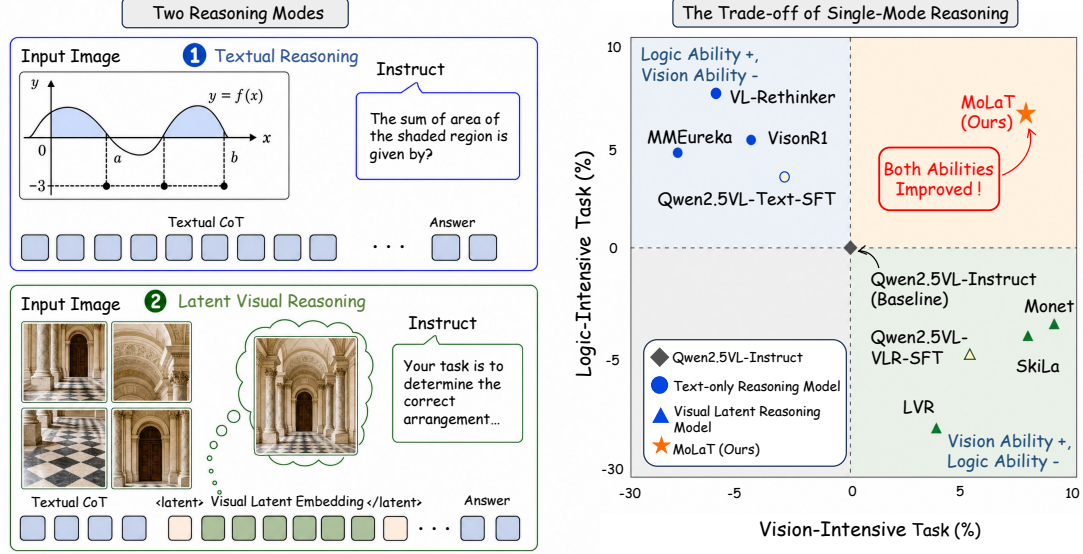


Figure 1: Illustration of the trade-off between latent visual reasoning and textual reasoning. LVR improves vision-intensive tasks but degrades logic-intensive performance, motivating adaptive mode selection.

question: *can explicit reward signals drive a model to learn an optimal mode-selection policy?*

To answer this, we propose **MoLaT (Mixture of Latent and Textual Reasoning)**, which unifies latent visual and textual reasoning within a single VLM and trains mode selection via explicit advantage signals. MoLaT adopts a two-stage paradigm. Stage one uses SFT as a cold start: unified mode prefixes (`<latent>` and `<text>`) serve as reasoning mode identifiers, and two types of reasoning CoT data are mixed within a single autoregressive framework, enabling the VLM to acquire LVR without forgetting its textual reasoning ability. Stage two introduces **mode-adaptive reinforcement learning** with two designs: (1) The **mode-adaptive advantage** decouples selection from content at the token level: mode-selection tokens receive a routing signal $\hat{S}(x)$ based on cross-mode expected-return differences, while content tokens retain the group-relative advantage. (2) The **entropy-modulated advantage weight** uses generation entropy as a difficulty proxy to modulate signal strength λ_{mode} : low-entropy problems transmit the routing signal at full strength; high-entropy problems indicate exceeding current VLM capability, so the unreliable signal is suppressed to avoid premature mode-selection convergence to suboptimal choices. This mechanism naturally induces an implicit curriculum from easy to hard problems.

Extensive experiments across fine-grained visual perception, spatial reasoning, mathematical problem solving, and hallucination benchmarks

show that MoLaT effectively learns both reasoning modes with adaptive selection capability, yielding consistent improvements across all settings.

2 Preliminaries

2.1 SFT for Latent Visual Reasoning

A VLM equipped with LVR capability must handle two heterogeneous output streams. For textual-reasoning steps, the model is optimized with the token-level cross-entropy loss $L_{\text{text}} = -\sum_t \log p_{\theta}(w_t | w_{<t}, x)$. For visual-reasoning steps, the model predicts continuous latent embeddings $\{z_1, \dots, z_K\}$ corresponding to intermediate visual thinking steps, trained via MSE loss $L_{\text{vis}} = \sum_k \|\hat{z}_k - z_k\|_2^2$. The overall objective linearly combines the two losses,

$$L = \lambda_{\text{text}} L_{\text{text}} + \lambda_{\text{vis}} L_{\text{vis}}, \quad (1)$$

where each sample contributes only to the modality it contains, enabling unified training over purely textual, purely visual, and interleaved traces.

2.2 RL for Latent Visual Reasoning

Applying RL to LVR is nontrivial because the visual reasoning process unfolds in a continuous hidden-state space, preventing a direct token-level probability ratio over latent steps. Prior work (Li et al., 2025b; Viveiros et al., 2026) caches the LVR hidden states $\tilde{h}_i^{\text{latent}} = \{h_i^{\text{latent},1}, \dots, h_i^{\text{latent},L}\}$ during rollout and replays them during policy-gradient updates, so that the importance ratio need

only be computed over textual tokens:

$$r_{i,t}(\theta) = \frac{\pi_{\theta}(y_{i,t} \mid x, \tilde{h}_i^{\text{latent}}, y_{i,<t})}{\pi_{\theta_{\text{old}}}(y_{i,t} \mid x, h_i^{\text{latent}}, y_{i,<t})}. \quad (2)$$

Combined with the clipped surrogate objective of GRPO (Shao et al., 2024b), the overall policy optimization maximizes the token-averaged clipped advantage $\min(r_{i,t} A_{j,t}, \text{clip}(r_{i,t}, 1-\epsilon, 1+\epsilon) A_{j,t})$ penalized by a KL term $\beta D_{\text{KL}}(\pi_{\theta} \parallel \pi_{\text{ref}})$, with $A_{j,t}$ the advantage, ϵ the clipping threshold, and β the KL regularization strength toward the reference policy π_{ref} . This formulation provides the computational foundation for our **mode-adaptive reinforcement learning** in Section 3.

2.3 Preliminary Study

While LVR has shown strong potential on vision-intensive tasks, a natural question arises: Does the textual reasoning capability become compromised once LVR is introduced? We compare text-reasoning VLMs such as MM-Eureka and Vision-R1 with LVR-style VLMs such as LVR and Monet on logic-intensive and vision-intensive tasks. LVR substantially improves vision-intensive performance, yet incurs a marked drop on logic-intensive tasks, revealing an inherent trade-off. To rule out that this gap originates from training-data distribution rather than the reasoning mode itself, we design a controlled experiment: On the same 8K-sample pool covering both task types, we use multimodal datasets with visual CoT pattern (to train LVR ability) and text-only CoT distilled from Gemini-3-pro (Gemini Team, Google, 2025) to train Qwen2.5-VL-7B (same dataset); see Appendix B for data details. Results in Figure 1 confirm the same trend: even with identical data distribution, the reasoning mode alone causes significant task-preference divergence.

These findings expose a fundamental limitation: purely textual reasoning excels at abstract logic but lacks fine-grained visual perception, whereas LVR strengthens visual understanding at the cost of symbolic reasoning. This directly motivates MoLaT, which unifies the two modes and routes each problem to its more effective reasoning mode.

3 Method

MoLaT unifies latent visual and textual reasoning within a single VLM and learns adaptive mode selection via reinforcement learning. As shown in Figure 2, MoLaT consists of two stages: an SFT

cold start that equips the VLM with dual-mode reasoning (Section 3.1), and *mode-adaptive reinforcement learning* that teaches the model to route each problem to its more effective mode (Section 3.2).

3.1 SFT with Dual-Mode Reasoning Data

The first stage establishes latent visual reasoning capability via supervised fine-tuning. Following prior work (Li et al., 2025b; Wang et al., 2025b), encoder-extracted embeddings of intermediate reasoning-step images serve as supervision targets, teaching the VLM to generate latent visual representations. Our SFT corpus covers three styles: (1) **purely visual CoT** (e.g., visual navigation, jigsaw puzzles), (2) **interleaved visual-textual CoT**, and (3) **purely textual CoT** (Appendix D). Each sample is prefixed with a mode identifier (`<latent>` or `<text>`) governing the output format: textual reasoning uses `<think>...</think>`, latent segments use `<|latent_start|>...<|latent_end|>`. These identifiers provide the structural basis for hierarchical advantage computation in Section 3.2.

Our preliminary study shows that interleaved visual-latent and textual CoT behaves like pure LVR: once latent tokens appear, the VLM’s textual reasoning capabilities are disrupted (Tong et al., 2025). We therefore treat any CoT containing latent tokens as the LVR mode, yielding a binary partition consistent with Section 3.2.

3.2 Mode-Adaptive Reinforcement Learning

Prior work provides a policy-gradient framework with latent-state replay for RL over LVR trajectories, but considers only a single mode without a selection mechanism. In MoLaT, the RL stage must jointly optimize mode selection (a discrete choice) and content generation (a sequential optimization). Standard GRPO computes rollout-level advantages and cannot differentiate these two levels. We propose **mode-adaptive reinforcement learning** with two complementary designs.¹

Mode-Adaptive Advantage. We formalize mode selection as a discrete action decision. Given input x (image and question), let $m \in \{\text{latent}, \text{text}\}$ denote the reasoning mode, and define the mode-conditional expected return as $U(m \mid x) \triangleq \mathbb{E}_{o \sim \pi_{\theta}(\cdot \mid x, m)}[r(o)]$. The mode-level advantage is $A_{\text{mode}}(x, m) = U(m \mid x) - V(x)$,

¹A lightweight format reward constraining each mode’s output structure (guaranteeing reliable mode identification) is used as a prerequisite; see Appendix E.

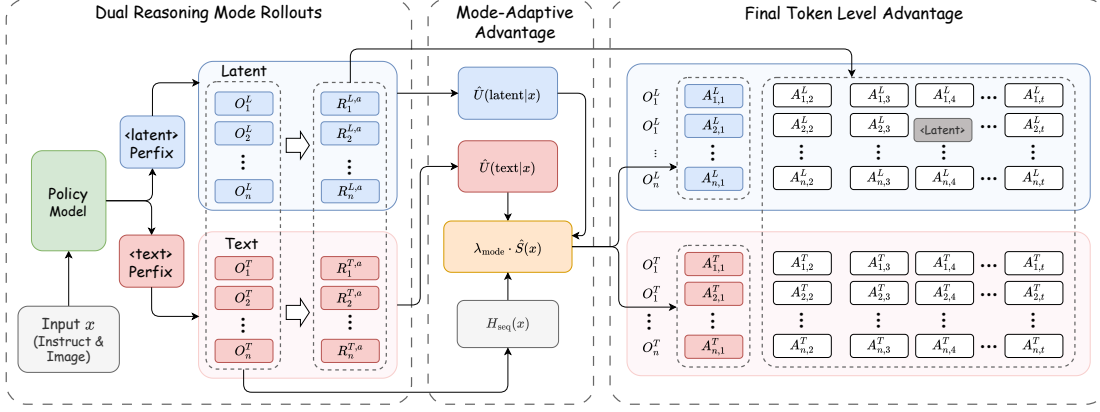


Figure 2: Overview of MoLaT. Stage 1: SFT cold start equips the VLM with dual-mode reasoning capability. Stage 2: Mode-adaptive reinforcement learning drives the model to learn optimal selection of reasoning modes via hierarchical advantage signals.

where $V(x)$ is the average return without conditioning on mode. In our binary setting, the baseline $V(x)$ cancels in the comparison (derivation in Appendix F), so we only need a single relative preference scalar $\hat{S}(x)$ that fully characterizes the gradient direction and magnitude of the routing decision. In practice, for each input x we fix the mode prefix to $\langle \text{latent} \rangle$ and $\langle \text{text} \rangle$ respectively, draw n rollouts per mode, and estimate the scalar via mean-variance normalization:

$$\hat{S}(x) = \frac{\hat{U}(\text{latent} | x) - \hat{U}(\text{text} | x)}{\sqrt{\hat{\sigma}_{\text{lat}}^2 + \hat{\sigma}_{\text{txt}}^2 + \epsilon}}, \quad (3)$$

where $\hat{U}(m | x) = \frac{1}{n} \sum_{k=1}^n r_k^{(m)}$. A positive $\hat{S}(x)$ indicates the input is better handled by LVR, and vice versa. Based on this signal, we define the advantage hierarchically at the token level:

$$A'_{j,t} = \begin{cases} +\lambda_m \cdot \hat{S}(x) & \text{if } o_{j,t} = \langle \text{latent} \rangle, \\ -\lambda_m \cdot \hat{S}(x) & \text{if } o_{j,t} = \langle \text{text} \rangle, \\ A_j^{\text{content}} & \text{otherwise,} \end{cases} \quad (4)$$

where λ_m is the mode-signal weight (whose adaptive schedule is defined below) and A_j^{content} is the standard group-relative advantage of rollout j . Mode-selection tokens receive a dedicated routing gradient derived from cross-mode comparison, while content tokens receive the standard within-mode quality gradient; the two are optimized independently under a single policy-gradient objective.

Entropy-Modulated Advantage Weight. Not all inputs are equally suitable for mode-selection learning. When a problem is within the VLM’s current capability, cross-mode comparisons are reliable and the routing signal should transmit at full

strength; when it far exceeds capability, the signal is noisy and may cause premature commitment to a suboptimal mode. We use sequence-level generation entropy as a difficulty proxy: high entropy indicates unreliable estimates that should be suppressed. Unlike token-level entropy for mitigating entropy collapse in RLVR (Yu et al., 2025; Cui et al., 2025), we use sequence-level entropy as a problem-level confidence measure.

Concretely, the token-level entropy at position t in rollout i is:

$$\mathcal{H}_t^i = -\sum_{v \in \mathcal{V}} \pi_\theta(v | q, a_{<t}^i) \log \pi_\theta(v | q, a_{<t}^i), \quad (5)$$

and the sequence-level entropy $H_{\text{seq}}(q)$ is the mean over all rollouts:

$$H_{\text{seq}}(q) = \frac{1}{N} \sum_{i=1}^N \frac{1}{T_i^{\text{txt}}} \sum_{t \in \mathcal{T}_i^{\text{txt}}} \mathcal{H}_t^i, \quad (6)$$

where N covers all rollouts under both modes and $\mathcal{T}_i^{\text{txt}}$ is the textual token positions in rollout i . Since LVR-mode latent tokens are not vocabulary-decoded, entropy is computed over textual tokens only. Following GTPO (Tan et al., 2025), we anchor bounds to π_{ref} ’s entropy distribution: ϵ_{low} and ϵ_{high} are the 10th and 90th percentiles of π_{ref} ’s sequence entropies, kept fixed throughout training. The mode-signal weight is:

$$\bar{H}(x) = \text{clip}(H_{\text{seq}}(x), \epsilon_{\text{low}}, \epsilon_{\text{high}}), \quad (7)$$

$$\lambda_m(x) = 1 - \frac{\bar{H}(x) - \epsilon_{\text{low}}}{\epsilon_{\text{high}} - \epsilon_{\text{low}}}, \quad (8)$$

where $\lambda_m(x) \in [0, 1]$. This induces an implicit curriculum: the VLM first establishes reliable mode-

selection on low-entropy problems, then progressively generalizes to harder ones as capability improves, driven entirely by learning progress rather than hand-crafted annotations.

4 Experiments

4.1 Experimental Setup

Implementation details. We validate MoLaT on Qwen2.5-VL-7B-Instruct (Bai et al., 2025b) and Qwen3-VL-8B (Bai et al., 2025a). The SFT stage uses $\sim 60K$ samples: purely visual CoT and interleaved visual-textual CoT data are drawn from Visual-CoT (Shao et al., 2024a), ThinkMorph (Gu et al., 2025), Zebra-CoT (Li et al., 2025a), and MathCanvas-Instruct (Shi et al., 2025), supplemented with textual CoT covering mathematical and logical reasoning to prevent forgetting of the textual mode. The RL stage trains on 30K curated samples balanced across vision-intensive and logic-intensive tasks. Data composition and hyperparameters are in the appendix.

Benchmarks. We evaluate on 10 benchmarks spanning two categories. **Vision-intensive:** V*Bench (Wu and Xie, 2024), BLINK (Tong et al., 2024), HR-Bench 4K/8K (Wang et al., 2025c), and MME-RealWorld-Lite (Zhang et al., 2024b), examining fine-grained visual search, spatial perception, high-resolution understanding, and real-world scene comprehension. **Logic-intensive:** MathVerse (Zhang et al., 2024a), MathVision (Wang et al., 2024a), MathVista (Lu et al., 2024), MMMU (Yue et al., 2024a), and MMMU-Pro (Yue et al., 2024b), covering mathematical and multi-discipline reasoning.

4.2 Main Results

Vision-intensive benchmarks. We compare MoLaT against three categories: (1) **Textual Reasoning** (Bai et al., 2025b; Huang et al., 2025; Meng et al., 2025); (2) **Latent Visual Reasoning** (Li et al., 2025b; Wang et al., 2025b; Tong et al., 2025, 2026); and (3) **Multimodal Agentic** methods invoking external visual tools (Zheng et al., 2025; Hong et al., 2025). As shown in Table 1, MoLaT achieves consistent gains on both base models, approaching 10 percentage points on HR-Bench and MME-RealWorld-Lite, surpassing all three baseline categories. Notably, MoLaT-Qwen3-VL-8B achieves state-of-the-art on all five vision-intensive benchmarks at comparable scale.

Logic-intensive benchmarks. This is the key

differentiating advantage of MoLaT over existing latent visual reasoning methods. As shown in Section 2, mainstream LVR methods suffer from significant degradation on logic-intensive tasks (e.g., LVR scores only ~ 31.5 on MathVerse). In contrast, MoLaT achieves stable improvements over the base model on all five benchmarks. On Qwen3-VL-8B, MoLaT matches or surpasses much larger models on most logic-intensive benchmarks (e.g., outperforming InternVL3.5-30B-A3B (Zhu et al., 2025) on MathVerse, MathVision, and MathVista, and VL-Rethinker-72B (Wang et al., 2025a) on all reported benchmarks), demonstrating that adaptive mode selection avoids degradation and actively strengthens logical reasoning.

Existing LVR methods exhibit a “robbing Peter to pay Paul” pattern: excelling on vision-intensive tasks at the cost of logical reasoning. The root cause is that continuous latent representations, when used as the sole reasoning space, encroach upon the capacity supporting symbolic reasoning. MoLaT resolves this through explicit mode-selection training: vision-intensive problems benefit from latent reasoning for fine-grained spatial modeling, while logic-intensive problems retain textual reasoning for maintaining deductive coherence. The two modes thereby form a complementary rather than competitive relationship.

4.3 Ablation Study

To validate each component of Mode-Adaptive RL, we ablate on Qwen2.5-VL-7B: (1) **w/o Entropy-Modulated Weight** fixes $\lambda_m(x) \equiv 1$, applying equal-strength mode-selection signals to all problems; (2) **w/o Mode-Adaptive Advantage** further removes the token-level hierarchical advantage, reverting to standard GRPO where all tokens share one group-relative advantage with only answer correctness as reward. We also include two single-mode baselines: **Text-SFT-RL** (SFT with text-only CoT) and **VLR-SFT-RL** (SFT with purely visual CoT and interleaved visual-textual CoT), both using vanilla GRPO since they lack dual modes. All variants use the same SFT data quantity as MoLaT, differing only in mode composition to rule out data-amount confounds.

Entropy-Modulated Weight. Fixing the weight to a constant drops ~ 1.6 points on vision-intensive and ~ 1.5 on logic-intensive tasks. This shows that uniform mode-preference signals propagate unreliable routing judgments from high-entropy problems, interfering with policy learning. The entropy-

Model	V*	BLINK	HR-Bench 4K	HR-Bench 8K	MME-RW-Lite
<i>Textual Reasoning Models</i>					
GPT-5.4 (OpenAI, 2026)	68.1	62.6	75.5	65.9	54.8
claude-opus4.5	60.2	66.9	71.4	64.9	56.3
Qwen2.5-VL-7B-Instruct (Bai et al., 2025b)	76.4	[†] 54.2	65.5	62.1	46.5
Qwen2.5-VL-32B-Instruct	80.6	55.1	69.3	63.6	–
Qwen3-VL-8B-Instruct (Bai et al., 2025a)	83.8	69.1	76.5	71.3	[†] 46.4
Qwen3-VL-8B-Thinking	77.5	60.9	72.4	68.1	–
LLaVA-OneVision-7B (Li et al., 2024)	75.4	53.0	63.0	59.8	48.5
Vision-R1 (Huang et al., 2025)	70.2	51.0	64.8	57.0	46.4
MM-Eureka (Meng et al., 2025)	51.3	38.5	–	–	–
<i>Latent Visual Reasoning Models</i>					
Monet (Wang et al., 2025b)	83.3	–	71.0	68.0	55.5
LVR (Li et al., 2025b)	81.7	52.5	69.6	66.1	49.1
SkiLa (Tong et al., 2025)	84.3	56.7	72.0	66.5	54.1
SwimBird (Tong et al., 2026)	85.5	–	[†] 78.4	74.9	–
<i>Multimodal Agentic Models</i>					
Pixel Reasoner (Su et al., 2025)	84.3	–	72.6	66.1	–
DeepEyes (Zheng et al., 2025)	83.3	–	73.2	69.5	54.3
DeepEyesV2 (Hong et al., 2025)	81.8	–	77.9	73.8	57.1
<i>Ours (Training Strategy Emphasizing Reasoning Mode Selection)</i>					
MoLaT-Qwen2.5-VL-7B	83.1	57.9	74.5	71.5	55.8
MoLaT-Qwen3-VL-8B	85.9	72.0	80.7	76.2	59.6

Table 1: Performance of MoLaT (based on Qwen2.5-VL-7B and Qwen3-VL-8B), whose training strategy emphasizes adaptive reasoning mode selection between latent visual reasoning and textual reasoning, on vision-intensive tasks. **Bold** scores represent the best performance among open-source VLMs at comparable scale ($\leq 10\text{B}$ parameters). [†] denotes results reproduced by us due to unreported scores in the original paper.

adaptive weight suppresses signals beyond current capability while reinforcing low-entropy ones, naturally forming an implicit curriculum that guides the routing policy along reliable convergence paths.

Mode-Adaptive Advantage. Further removing the token-level advantage produces the largest degradation among all variants: vision-intensive tasks drop over 2 points; logic-intensive scores regress substantially toward the base model. Without token-level decoupling, standard GRPO’s rollout-level advantage is shared across all tokens, diluting the gradient of the few mode-selection tokens among far more numerous content tokens. The model cannot develop effective mode-routing from correctness rewards alone, despite possessing dual-mode generation ability from SFT. Mode-Adaptive Advantage assigns dedicated routing gradients to mode-selection tokens while preserving content-token advantages, enabling independent optimization under one policy-gradient objective.

Single-mode comparison. Each single-mode variant achieves strong performance on its suited task type: Text-SFT-RL maintains strong math

reasoning while VLR-SFT-RL excels on vision-intensive tasks. Both outperform dual-mode SFT with vanilla GRPO, corroborating the trade-off in Section 2.3. MoLaT achieves high performance on both by encouraging appropriate mode selection: it matches Text-SFT-RL on math while surpassing VLR-SFT-RL on vision tasks. Combined with analysis in Section 4.4, this indicates that latent visual reasoning benefits from preserved textual capability, particularly during interleaved generation.

4.4 Analysis

Advantage-level vs. reward-level routing. MoLaT injects routing signals at the advantage level of mode-selection tokens. A natural alternative is to weight the cross-mode preference scalar $\hat{S}(x)$ into the per-rollout reward rather than the token-level advantage. The reward-based variant underperforms by ~ 1.8 points on vision-intensive and ~ 2.5 points on logic-intensive tasks. The mode signal at the reward level is diluted across all tokens’ advantage estimates after GRPO group normalization, preventing mode-selection tokens from receiv-

Model	Math-Reasoning			Multi-Disciplinary	
	MathVerse	MathVision	MathVista	MMMU	MMMU-Pro
Closed-Source Vision-Language Models					
OpenAI-o1 (OpenAI, 2024)	57.0	42.2	73.9	78.2	62.4
Claude-opus4.5	75.1	52.0	[†] 76.9	79.1	[†] 60.5
Open-Source Vision-Language Models					
Qwen2.5-VL-3B (Bai et al., 2025b)	–	21.2	62.3	[†] 51.2	31.6
Qwen2.5-VL-7B (Bai et al., 2025b)	45.6	25.1	68.2	[†] 54.3	36.9
Qwen3-VL-8B-Instruct (Bai et al., 2025a)	61.3	53.9	77.2	69.6	55.9
InternVL3.5-30B-A3B (Zhu et al., 2025)	60.4	55.7	80.9	75.6	–
VL-Rethinker-72B (Wang et al., 2025a)	61.7	43.9	80.3	68.8	53.9
MM-Eureka-32B (Meng et al., 2025)	56.5	–	74.8	[†] 55.4	[†] 40.6
SwimBird (Tong et al., 2026)	65.8	[†] 40.8	[†] 74.7	[†] 64.0	[†] 47.8
Ours (Training Strategy Emphasizing Reasoning Mode Selection)					
MoLaT-Qwen2.5-VL-7B	52.3	34.9	74.6	62.3	43.6
MoLaT-Qwen3-VL-8B	67.9	58.6	81.9	75.0	63.4

Table 2: Performance of MoLaT, whose training strategy emphasizes reasoning mode selection, on logic-intensive tasks including mathematical reasoning. **Bold** scores represent the best performance among open-source VLMs at comparable scale ($\leq 10B$ parameters). [†] denotes results reproduced by us due to unreported scores in the original paper.

Variant	Vision-Intensive			Logic-Intensive		
	V*	HR-Bench 4K	HR-Bench 8K	MathVerse	MathVision	MMMU
MoLaT-Qwen2.5-VL-7B	83.1	74.5	71.5	52.3	34.9	62.3
w/o Entropy-Mod. Weight	81.9	72.8	69.4	50.2	32.3	61.1
w/o Mode-Adaptive Adv.	80.7	70.5	67.1	49.4	29.6	59.2
Only-SFT	78.5	68.1	63.8	47.6	28.2	57.1
Text-SFT-RL	77.9	65.2	61.3	52.5	35.3	62.1
VLR-SFT-RL	82.4	73.8	70.2	41.8	23.4	48.5
Qwen2.5-VL-7B (baseline)	76.4	65.5	62.1	45.6	25.1	[†] 54.3

Table 3: Ablation study on the two core components of MoLaT: mode-adaptive advantage and entropy-modulated advantage weight. Text-SFT-RL and VLR-SFT-RL denote variants trained with only textual and only LVR reasoning modes for SFT and RL, respectively. All variants use the same total SFT data quantity.

ing independent gradient directions. In MoLaT, since mode-selection tokens already determine the reasoning mode (ensured by SFT format training and RL format reward), applying routing signals at their advantage level achieves targeted decoupling of mode routing from content optimization.

Comparison with curriculum learning. To verify the advantage of the implicit curriculum induced by Entropy-Modulated Advantage Weight over explicit curriculum learning, we design a curriculum baseline that divides RL data by difficulty and task diversity, training first on simpler, less diverse tasks and then on harder, more diverse ones. Curriculum learning improves over SFT-only but falls short of MoLaT, because data-partition-based difficulty criteria rely on human-defined standards and cannot reflect the current policy’s per-problem mode-selection confidence. Entropy-Modulated Advantage Weight dynamically senses reliability

via sequence entropy, making curriculum progression entirely policy-driven with finer granularity and no manual intervention.

Mode-selection behavior. MoLaT shows a strong preference for the latent mode on vision-intensive tasks and for the textual mode on logic-intensive tasks. This aligns with task characteristics: vision-intensive tasks bottleneck on spatial perception, where latent visual reasoning can model visual structure in continuous embedding space; logic-intensive tasks bottleneck on multi-step deduction, where textual reasoning better maintains logical coherence in discrete token space. Compared with vanilla GRPO, which exhibits weaker mode differentiation, Mode-Adaptive Advantage enables more decisive routing strategies via direct cross-mode gradient signals.

Variant	Vision-Intensive			Logic-Intensive		
	V*	HR-Bench 4K	HR-Bench 8K	MathVerse	MathVision	MMMU
MoLaT-Qwen2.5-VL-7B	83.1	74.5	71.5	52.3	34.9	62.3
w/ Reward-level routing	81.2	72.0	68.1	49.1	31.7	60.8
w/ Curriculum Learning	81.9	73.2	69.6	50.2	33.1	61.2

Table 4: Comparison of alternative design choices: reward-level mode-selection signal injection (weighting mode preference into the rollout reward rather than the token-level advantage) and explicit curriculum learning as a substitute for entropy-modulated advantage weighting.

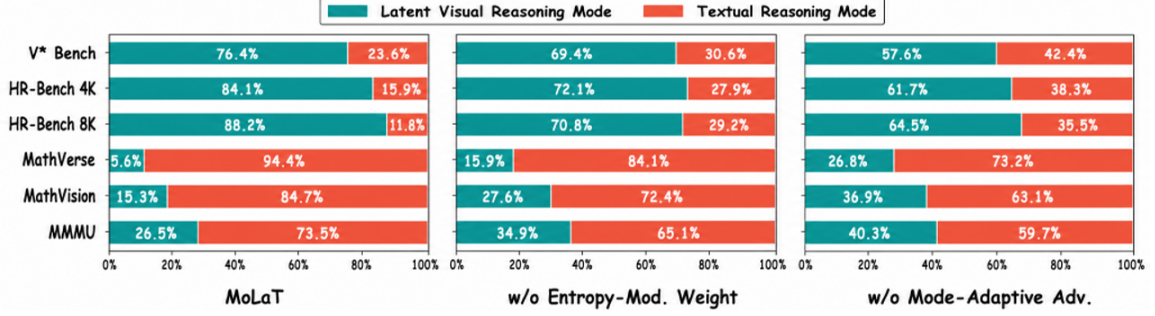


Figure 3: Mode-selection behavior of MoLaT across benchmarks. MoLaT preferentially selects the latent mode for vision-intensive tasks and the textual mode for logic-intensive tasks.

5 Related Work

Visual Reasoning. Early methods construct SFT datasets (Xu et al., 2024; Shao et al., 2024a) for multi-step textual reasoning. Recent work shifts to RL-based approaches (Huang et al., 2025; Meng et al., 2025) and self-reflection (Wang et al., 2025a). Another line augments reasoning with external visual tools: DeepEyes (Zheng et al., 2025; Hong et al., 2025) acquires visual evidence through interactive search and zooming; Pixel Reasoner (Su et al., 2025) incentivizes pixel-space reasoning via curiosity-driven RL. Despite progress, these methods remain confined to discrete textual space, with visual information as passive input or tool artifacts.

Thinking in Visual Latent Space. LVR (Li et al., 2025b) trains language models to generate latent states approximating query-relevant visual features, pioneering latent visual reasoning. Subsequent work advances along multiple axes: Monet (Wang et al., 2025b) proposes distillation-based SFT and VLPO; Latent Sketchpad (Zhang et al., 2025b) and SkiLa (Tong et al., 2025) achieve interleaved reasoning through internal visual sketchpads; Chain-of-Visual-Thought (Gao et al., 2025) constructs multi-step thought chains with continuous visual tokens. SwimBird (Tong et al., 2026) proposes adaptive switching among modes but relies on implicit SFT data guidance. The common limitation is that both latent reason-

ing and mode selection lack explicit RL feedback. MoLaT first drives routing policy learning via explicit mode reward signals.

Reinforcement Learning with Latent Tokens. GRPO (Shao et al., 2024b) and DeepSeek-R1 (Guo et al., 2025) establish group-relative policy optimization as the mainstream RL framework for reasoning. LVR (Li et al., 2025b) and LanteRn (Viveiros et al., 2026) adapt GRPO to latent reasoning via latent-state replay. Monet (Wang et al., 2025b) proposes VLPO to backpropagate through latent embeddings directly. Mirage (Yang et al., 2025) extends GRPO to interleaved text-latent trajectories. LaCoT (Sun et al., 2026) replaces GRPO with variational inference to improve generalization. All these works optimize reasoning quality within a single mode without addressing inter-mode routing. MoLaT targets a different objective: learning an explicit inter-mode routing policy through hierarchical advantage functions and entropy-modulated weighting.

6 Conclusion

We presented MoLaT, which unifies latent visual and textual reasoning in a single VLM and learns adaptive mode selection via reinforcement learning. Mode-adaptive advantage decouples mode-selection gradients from content-generation gradients at the token level, while entropy-modulated weighting suppresses unreliable routing signals

and induces an implicit curriculum. Experiments on ten benchmarks confirm consistent gains on both vision-intensive and logic-intensive tasks, validating adaptive mode selection as key to general-purpose multimodal reasoning.

Limitations

While MoLaT demonstrates consistent improvements across both vision-intensive and logic-intensive tasks, we identify several directions for future exploration. First, this work focuses on the binary mode partition, which is sufficient to validate the effectiveness of mode-adaptive reinforcement learning. A natural extension is to explore finer-grained mode decompositions, potentially unlocking further multimodal reasoning capabilities. Second, due to computational resource constraints, our evaluation focuses on VLMs at the 7–8B parameter scale.

Acknowledgments

We thank the anonymous reviewers for their valuable comments and suggestions. We are also grateful to our colleagues for helpful discussions and constructive feedback throughout this project.

References

- Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, and 45 others. 2025a. [Qwen3-vl technical report](#). *Preprint*, arXiv:2511.21631.
- Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibao Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, and 8 others. 2025b. [Qwen2.5-vl technical report](#). *arXiv preprint arXiv:2502.13923*.
- Vivek S. Borkar. 1997. Stochastic approximation with two time scales. *Systems & Control Letters*, 29(5):291–294.
- Ganqu Cui, Yuchen Zhang, Jiacheng Chen, Lifan Yuan, Zhi Wang, Yuxin Zuo, Haozhan Li, Yuchen Fan, Huayu Chen, Weize Chen, Zhiyuan Liu, Hao Peng, Lei Bai, Wanli Ouyang, Yu Cheng, Bowen Zhou, and Ning Ding. 2025. The entropy mechanism of reinforcement learning for reasoning language models. *arXiv preprint arXiv:2505.22617*.
- Shuai Dong, Siyuan Wang, Xingyu Liu, and Zhongyu Wei. 2025. Interleaved latent visual reasoning with selective perceptual modeling. *arXiv preprint arXiv:2512.05665*.
- Zhongxiang Gao, Siyuan Luo, Grace Luo, Trevor Darrell, and Roei Herzig. 2025. Chain-of-visual-thought: Teaching VLMs to see and think better with continuous visual tokens. *arXiv preprint arXiv:2511.19418*.
- Gemini Team, Google. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*.
- Jiawei Gu, Yunzhuo Hao, Huichen Will Wang, Linjie Li, Michael Qizhe Shieh, Yejin Choi, Ranjay Krishna, and Yu Cheng. 2025. ThinkMorph: Emergent properties in multimodal interleaved chain-of-thought reasoning. *arXiv preprint arXiv:2510.27492*.
- Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, and 175 others. 2025. [Deepseek-r1 incentivizes reasoning in llms through reinforcement learning](#). *Nature*, 645(8081):633–638.
- Jack Hong, Chenxiao Zhao, Chenglin Zhu, Weiheng Lu, Guohai Xu, and Xing Yu. 2025. DeepEyesV2: Toward agentic multimodal model. *arXiv preprint arXiv:2511.05271*.
- Yushi Hu, Weijia Shi, Xingyu Fu, Dan Roth, Mari Ostendorf, Luke Zettlemoyer, Noah A. Smith, and Ranjay Krishna. 2024. Visual sketchpad: Sketching as a visual chain of thought for multimodal language models. In *Advances in Neural Information Processing Systems (NeurIPS)*.
- Wenxuan Huang, Bohan Jia, Zijie Zhai, Shaosheng Cao, Zixian Ye, Fei Zhao, Yu Hu, and Shaohui Lin. 2025. Vision-R1: Incentivizing reasoning capability in multimodal large language models. *arXiv preprint arXiv:2503.06749*.
- Ang Li, Charles Wang, Deqing Fu, Kaiyu Yue, Zikui Cai, Wang Bill Zhu, Ollie Liu, Peng Guo, Willie Neiswanger, Furong Huang, Tom Goldstein, and Micah Goldblum. 2025a. [Zebra-cot: A dataset for interleaved vision language reasoning](#). *Preprint*, arXiv:2507.16746.
- Bangzheng Li, Ximeng Sun, Jiang Liu, Ze Wang, Jialian Wu, Xiaodong Yu, Hao Chen, Emad Barsoum, Muhao Chen, and Zicheng Liu. 2025b. Latent visual reasoning. *arXiv preprint arXiv:2509.24251*.
- Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. 2024. Llava-onevision: Easy visual task transfer.

- Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. 2024. MathVista: Evaluating mathematical reasoning of foundation models in visual contexts. In *International Conference on Learning Representations (ICLR)*.
- Fanqing Meng, Lingxiao Du, Zongkai Liu, Zhixiang Zhou, Quanfeng Lu, Daocheng Fu, Botian Shi, Wenhai Wang, Junjun He, Kaipeng Zhang, Ping Luo, Yu Qiao, Qiaosheng Zhang, and Wenqi Shao. 2025. MM-Eureka: Exploring visual aha moment with rule-based large-scale reinforcement learning. *arXiv preprint arXiv:2503.07365*.
- OpenAI. 2024. [Learning to reason with LLMs](#). *OpenAI Blog*.
- OpenAI. 2026. OpenAI GPT-5 system card. *arXiv preprint arXiv:2601.03267*.
- Hao Shao, Shengju Qian, Han Xiao, Guanglu Song, Zhuofan Zong, Letian Wang, Yu Liu, and Hongsheng Li. 2024a. Visual CoT: Advancing multi-modal language models with a comprehensive dataset and benchmark for chain-of-thought reasoning. In *Advances in Neural Information Processing Systems (NeurIPS)*.
- Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024b. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.
- Weikang Shi, Aldrich Yu, Rongyao Fang, Houxing Ren, Ke Wang, Aojun Zhou, Changyao Tian, Xinyu Fu, Yuxuan Hu, Zimu Lu, Linjiang Huang, Si Liu, Rui Liu, and Hongsheng Li. 2025. [Mathcanvas: Intrinsic visual chain-of-thought for multimodal mathematical reasoning](#). *Preprint*, arXiv:2510.14958.
- Alex Su, Haozhe Wang, Weimin Ren, Fangzhen Lin, and Wenhui Chen. 2025. Pixel reasoner: Incentivizing pixel-space reasoning with curiosity-driven reinforcement learning. *arXiv preprint arXiv:2505.15966*.
- Guohao Sun, Hang Hua, Jian Wang, Jiebo Luo, Soheil Dianat, Majid Rabbani, Raghuveer Rao, and Zhiqiang Tao. 2026. Latent chain-of-thought for visual reasoning. *Advances in neural information processing systems*, 38:103739–103762.
- Hongze Tan, Zihan Wang, Jianfei Pan, Jinghao Lin, Hao Wang, Yifan Wu, Tao Chen, Zhihang Zheng, Zhihao Tang, and Haihua Yang. 2025. GTPo and GRPO-S: Token and sequence-level reward shaping with policy entropy. *arXiv preprint arXiv:2508.04349*.
- Jintao Tong, Jiaqi Gu, Yujing Lou, Lubin Fan, Yixiong Zou, Yue Wu, Jieping Ye, and Ruixuan Li. 2025. Sketch-in-latents: Eliciting unified reasoning in MLLMs. *arXiv preprint arXiv:2512.16584*.
- Jintao Tong, Shilin Yan, Hongwei Xue, Xiaojun Tang, Kunyu Shi, Guannan Zhang, Ruixuan Li, and Yixiong Zou. 2026. SwimBird: Eliciting switchable reasoning mode in hybrid autoregressive MLLMs. *arXiv preprint arXiv:2602.06040*.
- Shengbang Tong, Zhuang Liu, Yuexiang Zhai, Yi Ma, Yann LeCun, and Saining Xie. 2024. Eyes wide shut? exploring the visual shortcomings of multimodal LLMs. *arXiv preprint arXiv:2401.06209*. CVPR 2024.
- André G. Viveiros, Nuno Gonçalves, Matthias Lindemann, and André F. T. Martins. 2026. LanteRn: Latent visual structured reasoning. *arXiv preprint arXiv:2603.25629*. ICLR 2026 Workshop on Multimodal Intelligence.
- Haozhe Wang, Chao Qu, Zuming Huang, Wei Chu, Fangzhen Lin, and Wenhui Chen. 2025a. VL-Rethinker: Incentivizing self-reflection of vision-language models with reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*. ArXiv preprint arXiv:2504.08837.
- Ke Wang, Juntong Pan, Weikang Shi, Zimu Lu, Mingjie Zhan, and Hongsheng Li. 2024a. Measuring multimodal mathematical reasoning with MATH-Vision dataset. *arXiv preprint arXiv:2402.14804*.
- Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. 2024b. Qwen2-vl: Enhancing vision-language model’s perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*.
- Qixun Wang, Yang Shi, Yifei Wang, Yuanxing Zhang, Pengfei Wan, Kun Gai, Xianghua Ying, and Yisen Wang. 2025b. Monet: Reasoning in latent visual space beyond images and language. *arXiv preprint arXiv:2511.21395*.
- Wenbin Wang, Liang Ding, Minyan Zeng, Xiabin Zhou, Li Shen, Yong Luo, and Dacheng Tao. 2025c. Divide, conquer and combine: A training-free framework for high-resolution image perception in multimodal large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*. ArXiv preprint arXiv:2408.15556.
- Linquan Wu, Tianxiang Jiang, Yifei Dong, Haoyu Yang, Fengji Zhang, Shichang Meng, Ai Xuan, Linqi Song, and Jacky Keung. 2026. LaViT: Aligning latent visual thoughts for multi-modal reasoning. *arXiv preprint arXiv:2601.10129*.
- Penghao Wu and Saining Xie. 2024. V*: Guided visual search as a core mechanism in multimodal LLMs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Guowei Xu, Peng Jin, Ziang Wu, Hao Li, Yibing Song, Lichao Sun, and Li Yuan. 2024. LLaVA-CoT: Let

- vision language models reason step-by-step. *arXiv preprint arXiv:2411.10440*.
- Zeyuan Yang, Xueyang Yu, Delin Chen, Maohao Shen, and Chuang Gan. 2025. Machine mental imagery: Empower multimodal reasoning with latent visual tokens. *arXiv preprint arXiv:2506.17218*.
- Qiyang Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Wang Zhang, and 16 others. 2025. [Dapo: An open-source llm reinforcement learning system at scale](#). *Preprint*, arXiv:2503.14476.
- Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, Cong Wei, Botao Yu, Ruibin Yuan, Renliang Sun, Ming Yin, Boyuan Zheng, Zhenzhu Yang, Yibo Liu, Wenhao Huang, and 3 others. 2024a. MMMU: A massive multi-discipline multimodal understanding and reasoning benchmark for expert AGI. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Xiang Yue, Tianyu Zheng, Yuansheng Zhang, Ge Zhang, Ruoqi Liu, Cong Wei, Yibo Liu, Yuansheng Ni, Renliang Sun, Wenhao Huang, Zhenzhu Yang, Yu Su, and Wenhao Chen. 2024b. MMMU-Pro: A more robust multi-discipline multimodal understanding benchmark. *arXiv preprint arXiv:2409.02813*.
- Chi Zhang, Haibo Qiu, Qiming Zhang, Zhixiong Zeng, Lin Ma, and Jing Zhang. 2025a. DeepSketcher: Internalizing visual manipulation for multimodal reasoning. *arXiv preprint arXiv:2509.25866*.
- Huanyu Zhang, Wenshan Wu, Chengzu Li, Ning Shang, Yan Xia, Yangyu Huang, Yifan Zhang, Li Dong, Zhang Zhang, Liang Wang, Tieniu Tan, and Furu Wei. 2025b. Latent sketchpad: Sketching visual thoughts to elicit multimodal reasoning in MLLMs. *arXiv preprint arXiv:2510.24514*.
- Renrui Zhang, Dongzhi Jiang, Yichi Zhang, Haokun Lin, Ziyu Guo, Pengshuo Qiu, Aojun Zhou, Pan Lu, Kai-Wei Chang, Peng Gao, and Hongsheng Li. 2024a. MathVerse: Does your multi-modal LLM truly see the diagrams in visual math problems? In *European Conference on Computer Vision (ECCV)*.
- Yi-Fan Zhang, Huanyu Zhang, Haochen Tian, Chaoyou Fu, Shuangqing Zhang, Junfei Wu, Feng Li, Kun Wang, Qingsong Wen, Zhang Zhang, Liang Wang, Rong Jin, and Tieniu Tan. 2024b. MME-RealWorld: Could your multimodal LLM challenge high-resolution real-world scenarios that are difficult for humans? *arXiv preprint arXiv:2408.13257*.
- Ziwei Zheng, Michael Yang, Jack Hong, Chenxiao Zhao, Guohai Xu, Le Yang, Chao Shen, and Xing Yu. 2025. DeepEyes: Incentivizing “thinking with images” via reinforcement learning. *arXiv preprint arXiv:2505.14362*.
- Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, Zhangwei Gao, Erfei Cui, Xuehui Wang, Yue Cao, Yangzhou Liu, Xingguang Wei, Hongjie Zhang, Haomin Wang, Weiye Xu, and 32 others. 2025. [InternV3: Exploring advanced training and test-time recipes for open-source multimodal models](#). *Preprint*, arXiv:2504.10479.

Table 5: Hyperparameters for the SFT stage.

Hyper-parameters	Value
Batch size	128
Learning rate	1×10^{-5}
Learning rate schedule	Cosine
Warmup ratio	0.05
Weight decay	1×10^{-2}
Gradient clipping	1.0
Optimizer	AdamW
BF16	True
Total samples	~60K

A Use Of AI Assistants

In accordance with the ACL Policy on AI Writing Assistance, we disclose that ChatGPT was used only for language-level polishing (grammar correction, minor rewording, and stylistic refinement) on portions of the manuscript text. AI tools were not used for research ideation, experimental design, implementation, data analysis, interpretation of results, or generation of scientific claims. All substantive content of this paper, including all findings and conclusions, was produced independently by the authors, who have reviewed and verified every AI-assisted edit and take full responsibility for the final manuscript.

B Implementation Details and Hyperparameters

We train MoLaT in two stages: supervised fine-tuning (SFT) followed by reinforcement learning (RL). Both stages use Qwen2.5-VL-7B and Qwen3-VL-8B as base models. All experiments are conducted on $8 \times \text{A100 80GB}$ GPUs using DeepSpeed ZeRO-3 for memory-efficient distributed training. Below we detail the hyperparameters for each stage.

SFT Stage. We fine-tune the base models on approximately 60K samples combining visual chain-of-thought and textual chain-of-thought data. The SFT stage teaches the model to produce reasoning in both latent and textual modes with appropriate format tokens. We use a cosine learning rate schedule with linear warmup, and apply gradient clipping with a max norm of 1.0 to stabilize training. The vision encoder is frozen during SFT to preserve pre-trained visual representations, while the language model backbone and the mode-selection head are fully trainable.

Latent Visual Reasoning Implementation. Following LVR (Li et al., 2025b), each latent reasoning step generates $K = 32$ continuous tokens of dimension $d = 3584$ (matching Qwen2.5-VL’s hidden dimension; $d = 3584$ for Qwen3-VL-8B as well). The K latent tokens are produced autoregressively: after the model generates the $\langle |\text{latent_start}| \rangle$ token, a linear projection head $W_{\text{lat}} \in \mathbb{R}^{d \times d}$ is applied to the current hidden state to produce the first latent token z_1 ; this token is then fed back into the transformer as the next input embedding, producing the hidden state for z_2 , and so on for K steps until $\langle |\text{latent_end}| \rangle$ is generated. During SFT training, supervision targets are SigLIP encoder embeddings of intermediate reasoning-step images, projected to model dimension via a frozen linear adapter; an MSE loss is applied at each of the K latent positions. During RL, we employ *latent-state replay*: the hidden states $\tilde{h}^{\text{latent}}$ produced during the rollout forward pass are cached and replayed during the policy-gradient backward pass, bypassing the non-differentiable latent generation for importance ratio computation (only textual tokens participate in $r_{i,t}(\theta)$ as described in Section 2.2). Consequently, RL does not directly update the latent-generation pathway; latent reasoning quality is established during SFT and remains fixed during RL, while RL optimizes *when* to invoke latent reasoning and the quality of textual content surrounding latent segments. The “mode-selection head” refers to the standard language model head predicting the $\langle \text{latent} \rangle$ or $\langle \text{text} \rangle$ token at position $t = 1$; no additional architectural parameters are introduced for mode selection beyond the special token embeddings added during SFT initialization.

RL Stage. We apply Group Relative Policy Optimization (GRPO) on approximately 30K samples. The RL stage encourages the model to select the appropriate reasoning mode and improve answer accuracy. For each prompt, we sample 10 responses in LVR mode and 10 responses in text-only mode (20 total per prompt) to form a group, compute the group-relative advantage, and apply mode-adaptive weighting before the policy gradient update. For single-mode ablation variants, we sample 20 responses in the available mode. The KL penalty coefficient β is tuned to prevent the policy from diverging too far from the SFT initialization, which we found crucial for maintaining format compliance. We use a smaller learning rate than SFT to ensure stable policy improvement.

Table 6: Hyperparameters for the RL stage.

Hyper-parameters	Value
Batch size	512
Learning rate	5×10^{-6}
Warmup ratio	0.05
Rollout num (LVR mode)	10
Rollout num (text-only mode)	10
Rollout num (single mode, ablation)	20
Rollout temperature	0.9
Rollout top- p	0.99
KL coefficient (β)	1×10^{-2}
Entropy weight (α)	1×10^{-2}
Format reward weight (λ_f)	0.2
Gradient clipping	1.0
Optimizer	AdamW
BF16	True
Total samples	$\sim 30K$

Inference. During inference, MoLaT generates the mode prefix token autoregressively as the first token of the response. No additional computational overhead is introduced beyond standard autoregressive decoding, since mode selection is a single-token decision. For latent mode, the model generates continuous latent tokens that are processed internally without being decoded to text; for textual mode, standard token-by-token generation proceeds as in conventional reasoning models.

Evaluation Protocol. All reported results are mean accuracy scores averaged over multiple independent evaluation runs with different random seeds for sampling-based decoding. For deterministic benchmarks (e.g., multiple-choice), we use greedy decoding to ensure reproducibility. For open-ended generation benchmarks, we report the mean over runs to account for sampling variance.

C Inference Prompt

We use the following system prompt during both RL training and inference to instruct the model on mode selection and output format. The prompt defines two reasoning modes—*latent visual reasoning* and *text-only reasoning*—and asks the model to adaptively choose the most appropriate one based on the nature of the question.

System Prompt for Dual-Mode Reasoning

You are a helpful assistant. The user asks a question related to an image, you need to solve it.

Please first think about the reasoning process in the mind and then provide the user with the answer. The reasoning process and answer are enclosed within `<think>` `</think>` and `<answer>` `</answer>` tags, respectively.

You have two modes of thinking, and you should adaptively choose the most appropriate one based on the nature of the question:

1. Latent Visual Reasoning: You may choose this mode when the question requires reasoning in the visual space, such as spatial relationship understanding, visual pattern recognition, geometric reasoning, visual puzzles, counting, visual navigation, or any task where step-by-step visual inspection of image regions is essential. In this mode, you can reason over the visual content, performing chain-of-thought reasoning anchored in the visual latent space (LVR). When using this mode, begin your response with the tag `<latent>`. The latent visual reasoning segments within the reasoning chain are enclosed by `<|latent_start|>...<|latent_end|>`.

2. Text-only Reasoning: Use this mode when the question can be answered through logical reasoning based primarily on textual interpretation of the image content (e.g., reading text/charts, knowledge-based QA, arithmetic, symbolic reasoning). When using this mode, begin your response with the tag `<text>`.

Choose the mode that best fits the task. The mode prefix (`<latent>` or `<text>`) is a routing decision that determines which reasoning pathway your response will follow.

This prompt serves three purposes. First, it defines the mode vocabulary (`<latent>` and `<text>`) that the format reward (Appendix E) validates. Second, it provides natural-language descriptions of when each mode is appropriate, giving the SFT-initialized policy a reasonable prior before RL fine-tuning. Third, it standardizes the output structure (`<think>...</think>` and `<answer>...</answer>`) to enable automated reward extraction during GRPO training.

D Data Details

We detail the data composition for both SFT and RL training stages. The SFT stage uses $\sim 60K$ samples to establish dual-mode reasoning capability, while the RL stage uses $\sim 30K$ curated samples to optimize mode selection through reward-driven

policy learning.

D.1 SFT Data Composition

Our SFT training data comprises approximately 60K samples drawn from three categories of chain-of-thought reasoning data, augmented with mode prefixes (<latent> or <text>) to enable mode-adaptive training. The data mixture is designed to expose the model to diverse reasoning patterns across both modalities during supervised learning.

Visual Chain-of-Thought Data. These samples require spatially grounded visual reasoning where intermediate steps reference image regions, including object localization, spatial relationship parsing, and visual attribute comparison. We source data from Visual-CoT (Shao et al., 2024a) and Zebra-CoT (Li et al., 2025a), which provide step-by-step visual reasoning traces with bounding box annotations and region-level descriptions. For these samples, the reasoning is wrapped with <|latent_start|>...<|latent_end|> tokens to train the latent reasoning pathway. The latent tokens encourage the model to perform visual computation in continuous embedding space rather than verbalizing each perceptual step.

Hybrid Visual-Textual CoT Data. These samples involve problems where visual perception feeds into structured symbolic reasoning (e.g., interpreting mathematical diagrams, reading charts, or parsing geometric figures). We use ThinkMorph and MathCanvas-Instruct (Shi et al., 2025), which provide reasoning traces that combine visual grounding with textual derivation steps. The hybrid format interleaves latent visual tokens with textual reasoning tokens, teaching the model to seamlessly transition between perceptual and symbolic processing within a single reasoning chain.

Textual Chain-of-Thought Data. For problems that are predominantly linguistic or logical—such as commonsense reasoning, arithmetic word problems, and knowledge-based QA—we include textual CoT samples formatted with <think>...</think> tags. These train the textual reasoning pathway for questions where visual latent computation offers no advantage. The textual CoT data is sourced from existing reasoning datasets and reformatted with our mode prefix convention.

Data Balancing. We balance the three categories to prevent mode bias during SFT. Empirically, we

Table 7: SFT data composition by source dataset. “V-T interleave” denotes vision-textual interleaved chain-of-thought.

Dataset	Samples	CoT Type
Visual-CoT	~5K	Vision only
Zebra-CoT	~5K	Vision only
Zebra-CoT	~10K	V-T interleave
ThinkMorph	~5K	V-T interleave
MathCanvas-Instruct	~5K	V-T interleave
OVR	~8K	Text only
OpenMMReasoner-SFT	~18K	Text only
CLEVR-Math	~2K	Text only
AI2D	~2K	Text only
Total	~60K	—

find that a roughly equal split between latent and textual samples (with hybrid samples counted toward both) yields the best downstream RL performance, as it initializes the mode-selection policy near uniform probability, allowing RL to shape preferences based on task-specific reward signals.

D.2 RL Data Composition

The RL stage uses approximately 30K curated samples for GRPO training. Unlike SFT, which teaches the model *how* to reason in each mode, the RL stage teaches the model *when* to select each mode by optimizing mode-selection decisions through reward signals. The RL data is designed with two key principles: (1) balanced coverage across vision-intensive and logic-intensive tasks to provide contrastive mode-selection signal, and (2) sufficient difficulty to ensure non-trivial mode routing (i.e., problems where mode choice meaningfully affects outcome).

Vision-Intensive Tasks. We include ~15K samples spanning fine-grained visual search, spatial reasoning, high-resolution image understanding, and real-world scene comprehension. These are drawn from Visual-CoT, VisualProbe, Zebra-CoT, DeepEyesV2_RL, MMStar, and MEGA. For these problems, latent visual reasoning typically achieves higher reward, providing positive signal for mode selection toward <latent>.

Logic-Intensive Tasks. We include ~15K samples covering mathematical reasoning, multi-step logical deduction, and multi-discipline knowledge tasks. These are drawn from OpenMMReasoner-RL, Honey, EMMA, and WeMath. For these problems, textual chain-of-thought typically achieves

Table 8: RL data composition by source dataset.

Dataset	Samples	Task Type
Visual-CoT	~1K	Vision-intensive
VisualProbe	~4K	Vision-intensive
Zebra-CoT	~2K	Vision-intensive
DeepEyesV2_RL	~6K	Vision-intensive
MMStar	~1K	Vision-intensive
MEGA	~1K	Vision-intensive
OpenMMReasoner-RL	~8K	Logic-intensive
Honey	~5K	Logic-intensive
EMMA	~1.5K	Logic-intensive
WeMath	~0.5K	Logic-intensive
Total	~30K	–

higher reward, providing positive signal for mode selection toward `<text>`.

Data Curation Criteria. We apply two filtering criteria to ensure data quality for mode-adaptive RL. First, we retain only **mode-discriminative** problems where the two reasoning modes produce meaningfully different reward distributions in pilot rollouts; problems where both modes achieve near-identical performance (e.g., trivial recognition tasks) are excluded, as they provide no mode-selection learning signal. Second, we ensure problems are **solvable but non-trivial**: we exclude problems that are either too easy (both modes achieve $>95\%$ accuracy, providing saturated reward) or too hard (both modes achieve $<5\%$ accuracy, providing no positive gradient signal), ensuring the RL stage operates in a productive learning regime.

Balanced Sampling Strategy. During RL training, we sample each mini-batch with approximately equal representation from vision-intensive and logic-intensive categories. This balance is critical for mode-adaptive learning: if one category dominates, the mode-selection policy may collapse to always choosing the mode favored by the majority category. Equal representation ensures that the cross-mode preference scalar $\hat{S}(x)$ takes both positive and negative values within each training batch, providing bidirectional gradient signals for the mode-routing policy.

Decontamination. We perform n -gram decontamination to ensure no overlap between training and evaluation data. For each evaluation benchmark, we compute 13-gram overlap between test questions and our SFT/RL training samples. Any

training sample with $\geq 70\%$ 13-gram overlap with any test instance is removed. The overlap rates before filtering are: V* Bench (0.2%), BLINK (0.0%), HR-Bench (0.0%), MathVerse (1.3%), MathVision (0.8%), MathVista (2.1%), MMMU (0.4%), MMMU-Pro (0.3%). After decontamination, fewer than 200 samples were removed from the combined SFT+RL training set. We verified that training on the decontaminated set produces scores within 0.3 points of the full set across all benchmarks, confirming that our results are not inflated by data leakage.

E Format Reward

During RL training, we employ a lightweight format reward r_{format} that ensures the model’s output is parseable and mode-identifiable. This reward is orthogonal to the correctness reward and serves to maintain structural validity of the generated reasoning throughout policy optimization.

Motivation. Without explicit format enforcement, we observe that RL training can degrade output structure: the model may omit mode prefixes, produce malformed reasoning blocks, or mix tokens from both modes in a single response. The format reward provides a consistent training signal that prevents such degeneration while imposing minimal constraint on reasoning content.

Format Checks. Specifically, the format reward verifies three conditions. First, **mode prefix presence**: the response must begin with exactly one of `<latent>` or `<text>`, indicating the chosen reasoning mode; responses that begin with neither or both tokens receive zero format reward. Second, **reasoning block validity**: if the mode is textual, the reasoning must be enclosed in matched `<think>...</think>` tags; if the mode is latent, the reasoning must be enclosed in matched `<|latent_start|>...<|latent_end|>` tags; nesting, duplication, or missing closure all constitute violations. Third, **answer block presence**: the response must contain a final answer in the expected format (e.g., `\boxed{}` for math or a letter choice for multiple-choice) after the reasoning block, ensuring the model commits to a parseable answer rather than trailing off.

Reward Definition. The format reward is defined as:

$$r_{\text{format}}(y) = \begin{cases} 1 & \text{if all checks pass,} \\ 0 & \text{otherwise.} \end{cases} \quad (9)$$

The total reward used in GRPO combines correctness and format:

$$r(x, y) = r_{\text{correct}}(x, y) + \lambda_f \cdot r_{\text{format}}(y), \quad (10)$$

where $\lambda_f = 0.2$ is the format reward weight. We set λ_f small enough that format compliance alone cannot dominate the reward signal, but large enough to maintain near-perfect format adherence (>99%) throughout training.

Effect on Training Stability. In ablation experiments, removing the format reward leads to a gradual increase in malformed outputs (reaching $\sim 8\%$ by the end of RL training), which corrupts advantage estimation for the mode-adaptive component. The format reward thus serves a dual purpose: it directly maintains output parseability and indirectly stabilizes mode-adaptive learning by ensuring clean mode labels for advantage computation.

F Derivation of Mode-Adaptive Advantage

We provide a complete derivation of the mode-adaptive advantage mechanism in MoLaT. We show why the value function $V(x)$ cancels in the binary mode selection setting, yielding a simplified advantage that directly compares mode-specific expected returns, and explain the two-timescale optimization interpretation of the entropy-modulated weight.

F.1 Mode Selection as a Structured Action

Setup. Consider a prompt x (image + question) for which the policy selects mode $m \in \{\text{latent}, \text{text}\}$ with probability $\pi(m | x)$. The key observation is that reward r is not directly determined by the mode choice m , but rather by the final output sequence y generated conditioned on m . We therefore define the state-action value function (mode Q-value) as:

$$Q(x, m) \triangleq \mathbb{E}_{y \sim \pi(\cdot | x, m)}[r(x, y)], \quad (11)$$

i.e., the expected reward when the model commits to mode m and then generates a response y according to the mode-conditional policy. This is

equivalent to $U(m | x)$ in the main text. The value function marginalizes over modes:

$$V(x) = \sum_m \pi(m | x) \cdot Q(x, m), \quad (12)$$

representing the expected reward when the model selects modes according to its current policy without external intervention.

F.2 Advantage Decomposition and $V(x)$ Cancellation

Advantage Decomposition. The advantage of selecting mode m follows the standard RL definition:

$$\begin{aligned} A(x, m) &= Q(x, m) - V(x) \\ &= Q(x, m) \\ &\quad - \sum_{m'} \pi(m' | x) Q(x, m'). \end{aligned} \quad (13)$$

In the binary setting, let $p = \pi(\text{latent} | x)$. Expanding the value function:

$$V(x) = p \cdot Q(x, \text{lat}) + (1-p) \cdot Q(x, \text{txt}). \quad (14)$$

Substituting into the advantage for the latent mode:

$$\begin{aligned} A(x, \text{lat}) &= Q(x, \text{lat}) - V(x) \\ &= Q(x, \text{lat}) - p \cdot Q(x, \text{lat}) \\ &\quad - (1-p) \cdot Q(x, \text{txt}) \\ &= (1-p) [Q(x, \text{lat}) - Q(x, \text{txt})]. \end{aligned} \quad (15)$$

Similarly, for the textual mode:

$$A(x, \text{txt}) = p \cdot [Q(x, \text{txt}) - Q(x, \text{lat})]. \quad (16)$$

Emergence of the Cross-Mode Preference Scalar.

In the binary mode setting, the advantage has only one degree of freedom. Taking the difference:

$$\begin{aligned} A(x, \text{lat}) - A(x, \text{txt}) &= Q(x, \text{lat}) - Q(x, \text{txt}), \end{aligned} \quad (17)$$

where the baseline $V(x)$ cancels. This motivates defining the *cross-mode preference scalar*:

$$\begin{aligned} S(x) &\triangleq Q(x, \text{lat}) - Q(x, \text{txt}) \\ &= U(\text{lat} | x) - U(\text{txt} | x), \end{aligned} \quad (18)$$

which fully characterizes the mode preference direction and magnitude. When $S(x) > 0$, latent mode is preferred; when $S(x) < 0$, textual mode is preferred.

Interpretation and Design Choice. The theoretical advantage $A(x, \text{lat}) = (1-p) S(x)$ contains the factor $(1-p) = 1 - \pi(\text{latent} \mid x)$, which shrinks the gradient when the policy is already confident. In Eq. (22) we deliberately use the constant-weight form $\pm \lambda_m(x) \hat{S}(x)$ instead. We do not claim this constitutes an unbiased policy-gradient estimator for the mode token; rather, it is a practical routing objective that directly maximizes the cross-mode return difference signal for the single routing token. The bias introduced by omitting $(1-p)$ is bounded by the KL constraint toward π_{ref} , which prevents extreme mode commitment. The rationale for this design choice is that retaining $(1-p)$ would suppress corrective gradients for committed-but-wrong routing, whereas entropy modulation $\lambda_m(x)$ already handles signal suppression based on problem-level reliability rather than policy confidence.

F.3 Empirical Estimation via GRPO

Forced-Mode Balanced Rollout Estimation. In practice, we estimate $S(x)$ using *forced-mode balanced rollouts*. For each prompt x , we generate two groups of rollouts: n rollouts with the mode prefix forced to $\langle \text{latent} \rangle$ (group $\mathcal{G}_{\text{lat}}$) and n rollouts with the mode prefix forced to $\langle \text{text} \rangle$ (group $\mathcal{G}_{\text{txt}}$), yielding $G = 2n$ total rollouts per prompt. Content generation after the forced prefix proceeds autoregressively under the current policy π_θ . This forced-mode design ensures balanced estimation of both mode-conditional returns regardless of the current policy’s mode preference, eliminating variance from unbalanced mode sampling. The empirical estimates are:

$$\hat{U}(\text{lat} \mid x) = \frac{1}{|\mathcal{G}_{\text{lat}}|} \sum_{k \in \mathcal{G}_{\text{lat}}} r(x, y_k), \quad (19)$$

$$\hat{U}(\text{txt} \mid x) = \frac{1}{|\mathcal{G}_{\text{txt}}|} \sum_{k \in \mathcal{G}_{\text{txt}}} r(x, y_k), \quad (20)$$

yielding $\hat{S}(x) = \hat{U}(\text{lat} \mid x) - \hat{U}(\text{txt} \mid x)$.

Cancellation of $V(x)$. Since both forced-mode groups share the same prompt x , the group mean reward $\bar{r} = \frac{1}{G} \sum_j r(x, y_j)$ serves as an empirical estimate of $V(x)$. When comparing two responses y_i (from $\mathcal{G}_{\text{lat}}$) and y_j (from $\mathcal{G}_{\text{txt}}$) within the same group, the shared baseline cancels:

$$\begin{aligned} \hat{A}(x, y_i) - \hat{A}(x, y_j) \\ = r(x, y_i) - r(x, y_j). \end{aligned} \quad (21)$$

This enables the policy gradient to directly reinforce the mode achieving higher reward without requiring a learned value network as a critic—a significant simplification over actor-critic methods.

Token-Level Advantage Assignment. Based on the cross-mode preference scalar, we assign dedicated advantages to mode-selection tokens while preserving standard GRPO advantages for content tokens:

$$A'_{j,t} = \begin{cases} +\lambda_m(x) \cdot \hat{S}(x) & \text{if } o_{j,t} = \langle \text{latent} \rangle \\ -\lambda_m(x) \cdot \hat{S}(x) & \text{if } o_{j,t} = \langle \text{text} \rangle \\ A_j^{\text{content}} & \text{otherwise} \end{cases} \quad (22)$$

This decouples mode-routing optimization from content optimization: mode-selection tokens receive a relative advantage scalar $S(x)$ that directly encodes cross-mode preference, while content tokens retain the standard rollout-level advantage for improving reasoning quality.

Content-Token Advantage Computation. To ensure true decoupling between routing and content optimization, the content-token advantage A_j^{content} is computed via group normalization *within each forced-mode group* separately. That is, for rollouts in $\mathcal{G}_{\text{lat}}$, the content advantage is $A_j^{\text{content}} = r(x, y_j) - \bar{r}_{\text{lat}}$ where $\bar{r}_{\text{lat}} = \frac{1}{n} \sum_{k \in \mathcal{G}_{\text{lat}}} r(x, y_k)$; analogously for $\mathcal{G}_{\text{txt}}$. This within-mode normalization ensures that content tokens are optimized for reasoning quality relative to other rollouts *of the same mode*, preventing cross-mode reward differences from leaking into content-token gradients.

Numerical Stability of $\hat{S}(x)$. The denominator of $\hat{S}(x)$ uses $\epsilon = 10^{-8}$ for numerical stability. Additionally, $\hat{S}(x)$ is clipped to $[-3, 3]$ before being used as the mode-token advantage, preventing excessively large routing signals when one mode achieves uniformly high reward and the other uniformly low reward (yielding near-zero within-mode variance).

F.4 Entropy-Modulated Advantage Weight

The entropy-modulated weight $\lambda_m(x)$ in Eq. (22) controls the strength of mode-selection gradients based on the policy’s generation uncertainty. As defined in Section 3.2, the weight is computed from the sequence-level entropy $H_{\text{seq}}(x)$ —the mean token-level vocabulary entropy over all rollouts—clipped and normalized to $[0, 1]$ using the 10th/90th

percentile bounds of π_{ref} 's entropy distribution:

$$\lambda_m(x) = 1 - \frac{\tilde{H}(x) - \epsilon_{\text{low}}}{\epsilon_{\text{high}} - \epsilon_{\text{low}}}. \quad (23)$$

When $\lambda_m \approx 1$ (low entropy, confident generation), mode-selection gradients transmit at full strength because the model's uncertainty is low and the mode preference signal is reliable. When $\lambda_m \approx 0$ (high entropy, uncertain generation), mode-selection gradients are suppressed because the problem exceeds current capability and premature mode commitment would inject noise.

F.5 Two-Timescale Optimization Interpretation

The entropy-modulated weight induces a natural two-timescale optimization dynamic:

Routing vs. Content. Mode-selection (routing) is a low-dimensional, binary decision with strong deterministic structure, while content generation is high-dimensional and non-convex. The annealing behavior of $\lambda_m(x)$ approximates a two-timescale update rule (Borkar, 1997): routing converges faster (receiving strong signals when entropy is low) while content optimization proceeds on a slower timescale across all prompts. Formally, for prompts where the mode preference is clear early in training (low $\tilde{H}$), the routing policy receives amplified gradients and converges rapidly; for ambiguous prompts (high $\tilde{H}$), routing signals are suppressed until content-level improvement reduces entropy sufficiently to make mode differences distinguishable.

Implicit Curriculum Effect. This creates an implicit curriculum without explicit data partitioning: easy problems (low entropy, clear mode preference) are prioritized for mode-routing learning, while hard problems (high entropy, ambiguous mode preference) are deferred until the policy accumulates sufficient evidence. Unlike explicit curriculum learning which requires human-defined difficulty criteria, the entropy-modulated weight is entirely policy-driven, adapting automatically as model capability evolves during training.

Prevention of Mode Collapse. Even though the implementation omits the $(1-p)$ factor, mode collapse is prevented by two complementary mechanisms: (1) the KL penalty toward π_{ref} (which has balanced mode probabilities from SFT) resists extreme mode commitment; (2) the entropy-modulated weight down-regulates routing signals

on problems where the current policy has not yet accumulated sufficient reward evidence to justify a confident mode choice.

Computational Cost. MoLaT's RL stage requires $2n$ rollouts per prompt ($n=10$ per mode, 20 total), roughly doubling the rollout cost compared to standard single-mode GRPO with 10 rollouts. On $8 \times \text{A100 80GB GPUs}$, the RL stage completes in approximately 48 hours for 30K training prompts. At inference time, mode selection adds negligible overhead (a single token decision); however, selecting latent mode incurs additional computation from generating $K=32$ continuous latent tokens per reasoning step compared to text-only mode.

G Supplementary Experiments

This section provides supplementary experimental results that complement the main paper, including: (1) the full preliminary study motivating the reasoning mode trade-off, (2) extended ablation and analysis on Qwen3-VL-8B, and (3) learning dynamics analysis across domains.

G.1 Reasoning Mode Trade-off: Full Preliminary Study

Table 9 presents the complete results of the preliminary study introduced in Section 2.3 of the main paper. We compare three categories of methods on both vision-intensive (V^* Bench, MME-RealWorld-Lite) and logic-intensive (MathVerse, MathVista) benchmarks: (1) textual reasoning models that apply RL with text-only chain-of-thought; (2) latent visual reasoning (LVR) models that reason in continuous visual embedding space; and (3) controlled single-mode SFT variants on Qwen2.5-VL-7B using identical data distribution.

Key observations. The results reveal a stark and consistent trade-off. **Textual reasoning models** (MM-Eureka, Vision-R1, VL-Rethinker) improve MathVerse by +4.5–6.8 and MathVista by +4.8–6.7 over the baseline, but degrade significantly on V^* Bench (−6.2 to −25.1) and MME-RW-Lite (−0.1 to −8.0). The most extreme case is MM-Eureka, which drops 25 points on V^* while gaining only 5 on MathVerse. **LVR models** (LVR, Monet, SkiLa) improve V^* Bench by +5.3–7.9 and MME-RW-Lite by +2.6–9.0, but suffer devastating drops on MathVerse (−12.4 to −15.0) and MathVista (−4.9 to −39.3). LVR's MathVista score of 28.9 represents

Model	Vision-Intensive		Logic-Intensive	
	V* Bench	MME-RW-Lite	MathVerse	MathVista
<i>Baseline</i>				
Qwen2.5-VL-7B (Bai et al., 2025b)	76.4	46.5	45.6	68.2
<i>Textual Reasoning Models (text-only CoT + RL)</i>				
MM-Eureka (Meng et al., 2025)	51.3	38.5	51.9	73.0
Vision-R1 (Huang et al., 2025)	70.2	46.4	52.4	73.5
VL-Rethinker (Wang et al., 2025a)	64.9	39.6	54.2	74.9
<i>Latent Visual Reasoning Models</i>				
LVR (Li et al., 2025b)	81.7	49.1	32.4	28.9
Monet (Wang et al., 2025b)	83.3	55.5	35.0	63.3
SkiLa (Tong et al., 2025)	84.3	54.1	34.2	55.1
<i>Controlled Single-Mode SFT on Qwen2.5-VL-7B (same data distribution)</i>				
Qwen2.5-VL-7B (Text Reasoning SFT)	73.5	44.7	49.8	71.3
Qwen2.5-VL-7B (Latent Reasoning SFT)	79.5	50.3	31.6	60.2

Table 9: Full preliminary study results revealing the reasoning mode trade-off. Textual reasoning models improve logic-intensive performance but degrade or stagnate on vision-intensive tasks. LVR models substantially improve vision-intensive performance but suffer catastrophic degradation on logic-intensive tasks (e.g., LVR drops to 32.4 on MathVerse, 28.9 on MathVista). The controlled SFT experiment on identical data confirms the trade-off originates from the reasoning mode itself rather than data distribution.

a -39.3 point collapse, indicating that continuous latent representations actively interfere with symbolic reasoning. The **controlled SFT experiment** isolates the reasoning mode as the causal factor: with the same 8K training samples, Latent Reasoning SFT improves V* by $+3.1$ but drops MathVerse by -15.8 , while Text Reasoning SFT drops V* by -2.9 but improves MathVerse by $+2.4$. This rules out data-distribution confounds and confirms that *the reasoning mode itself* causes task-preference divergence.

Implication for MoLaT. These findings demonstrate that neither single reasoning mode is universally optimal. The trade-off is fundamental rather than incidental: latent visual reasoning excels at fine-grained spatial modeling but encroaches upon symbolic reasoning capacity, while textual reasoning maintains deductive coherence but cannot leverage visual structure. This motivates MoLaT’s mode-adaptive design, which routes each problem to its more effective reasoning mode via learned mode selection, achieving complementary rather than competitive utilization of both capabilities.

G.2 Extended Ablation Study on Qwen3-VL-8B

In the main paper, we present ablation studies on Qwen2.5-VL-7B. Here we provide the complete ablation and analysis results on the stronger Qwen3-

VL-8B base model, demonstrating that MoLaT’s design choices remain effective across different model capabilities.

Observations. The ablation results on Qwen3-VL-8B are consistent with Qwen2.5-VL-7B in the main paper, with several noteworthy differences. On **vision-intensive tasks**, the relative gaps are smaller because Qwen3-VL-8B’s stronger visual backbone means that even without mode-adaptive training, the baseline achieves higher vision scores; nevertheless, MoLaT still improves by $+2.1/+4.2/+4.9$ on V*/HR-4K/HR-8K over the baseline. On **logic-intensive tasks**, the absolute gains are larger: compared to the base model, MoLaT achieves $+6.6/+5.7$ on MathVerse/MathVision, confirming that mode-adaptive RL effectively preserves and enhances textual reasoning on a stronger backbone. The **single-mode trade-off persists**: Text-SFT-RL achieves 68.1 on MathVerse (even surpassing MoLaT by 0.2 on this benchmark) but drops significantly on vision-intensive tasks (-4.9 on HR-8K vs. baseline); VLR-SFT-RL achieves 80.4 on HR-4K but catastrophically degrades on MathVerse (54.1, -7.2 vs. baseline). MoLaT achieves balanced improvements across both categories.

Variant	Vision-Intensive			Logic-Intensive	
	V* Bench	HR-Bench 4K	HR-Bench 8K	MathVerse	MathVision
MoLaT-Qwen3-VL-8B	85.9	80.7	76.2	67.9	58.6
w/o Entropy-Mod. Advantage Weight (LVR RL)	85.0	79.1	74.6	65.8	57.7
w/o Mode-Adaptive Advantage	84.4	78.4	73.6	64.9	55.8
Only-SFT	83.5	77.9	72.8	63.4	54.4
Text-SFT-RL	84.5	75.8	69.5	68.1	59.8
VLR-SFT-RL	85.3	80.4	75.0	54.1	47.6
Qwen3-VL-8B (baseline)	83.8	76.5	71.3	61.3	53.9

Table 10: Ablation study on Qwen3-VL-8B. The same component hierarchy observed on Qwen2.5-VL-7B holds: Mode-Adaptive Advantage contributes the largest gain, with Entropy-Modulated Weight providing additional improvement. Single-mode baselines (Text-SFT-RL and VLR-SFT-RL) each dominate on their respective task types but suffer on the other, confirming the necessity of adaptive mode selection.

Variant	Vision-Intensive			Logic-Intensive	
	V* Bench	HR-Bench 4K	HR-Bench 8K	MathVerse	MathVision
MoLaT-Qwen3-VL-8B	85.9	80.7	76.2	67.9	58.6
replace Mode-Adv. with Mode-Reward	85.0	78.7	74.5	64.7	58.0
replace Entropy-Mod. Weight with Explicit C.L.	85.3	79.8	75.5	65.8	59.0

Table 11: Comparison of alternative design choices on Qwen3-VL-8B. "Mode-Adv." stands for Mode-Adaptive Advantage. "Mode-Reward" denotes injecting the mode-selection signal at the reward level. "Explicit C.L." denotes replacing entropy-modulated advantage weighting with explicit curriculum learning.

G.3 Advantage-Level vs. Reward-Level Routing on Qwen3-VL-8B

Mode-Adaptive Advantage outperforms Mode-Reward. When mode-selection signals are injected at the reward level (Mode-Reward, i.e., weighting $\hat{S}(x)$ into per-rollout reward before GRPO group normalization), performance drops by $-0.9/-2.0/-1.7$ on V*/HR-4K/HR-8K and $-2.2/-1.6$ on MathVerse/MathVision. The degradation is particularly pronounced on logic-intensive tasks, where mode-selection tokens constitute a negligible fraction of total tokens. Reward-level injection dilutes the mode signal across all content tokens during advantage normalization, making it difficult for the few mode-selection tokens to receive directionally meaningful gradients.

Entropy-Modulated Weight outperforms Explicit C.L. The Explicit C.L. baseline (training first on simpler, less diverse tasks with clear mode preferences, then harder ambiguous tasks) narrows the gap to MoLaT but cannot match it ($-0.6/-0.9/-0.7$ on vision; $-1.1/-0.6$ on logic). The fundamental limitation of explicit curriculum learning is its reliance on pre-defined difficulty criteria that cannot adapt to the evolving policy. In contrast, entropy-modulated weighting dynamically senses per-problem reliability through the

model’s own generation uncertainty, providing a finer-grained, policy-adaptive implicit curriculum.

G.4 Learning Dynamics Across Domains

We present additional analysis of learning dynamics across different task categories beyond those shown in the main paper. These observations illustrate how the mode selection probability $\pi(\text{latent } | \ x)$ evolves during RL training.

General Trends. Across all domains, we observe a consistent three-phase pattern in mode selection evolution. In the **exploration phase** (early training), the mode selection probability remains near 0.5 for most samples, reflecting the balanced SFT initialization; the model explores both modes with similar frequency. In the **differentiation phase** (mid training), mode preferences begin to diverge across task types—vision-intensive tasks shift toward latent mode while logic-intensive tasks shift toward textual mode; this phase exhibits the highest gradient variance. In the **convergence phase** (late training), mode preferences stabilize with high confidence ($p > 0.8$ or $p < 0.2$ for most samples), and the entropy-modulated weight suppresses remaining noise.

Domain-Specific Observations. For **spatial reasoning** tasks (e.g., object counting, relative posi-

tion), the latent mode preference reaches approximately 85% by convergence, indicating strong reliance on visual computation. For **chart and graph understanding**, the model converges to a more balanced split ($\sim 65\%$ latent), reflecting that these tasks require both visual parsing and numerical reasoning. For **science diagram** interpretation, latent preference varies by sub-type: circuit diagrams favor latent mode strongly ($\sim 90\%$) while biology classification diagrams show weaker preference ($\sim 60\%$), likely because the latter relies more on textual knowledge retrieval.

Failure Cases. We also observe a small subset of samples ($< 5\%$) where mode selection oscillates throughout training without converging. These typically involve ambiguous tasks where both modes achieve similar reward (e.g., simple visual QA answerable through either route). The entropy-modulated weight effectively down-weights these samples, preventing them from destabilizing the overall policy.

The learning dynamics consistently show that the model learns to allocate latent reasoning to visually intensive tasks while preferring textual reasoning for problems that are predominantly linguistic. This emergent specialization validates the mode-adaptive design of MoLaT and confirms that explicit reward-driven mode selection outperforms implicit data-driven routing.

G.5 Case Study

We present qualitative examples illustrating how MoLaT’s adaptive mode selection leads to correct answers where single-mode reasoning fails.

These examples demonstrate that neither reasoning mode is universally superior: vision-intensive tasks requiring fine-grained spatial perception (OCR, pattern recognition) benefit from latent visual reasoning, while logic-intensive tasks requiring symbolic computation (geometry proofs, financial calculations) benefit from textual reasoning. MoLaT’s learned mode-selection policy correctly routes each problem to its more effective mode.

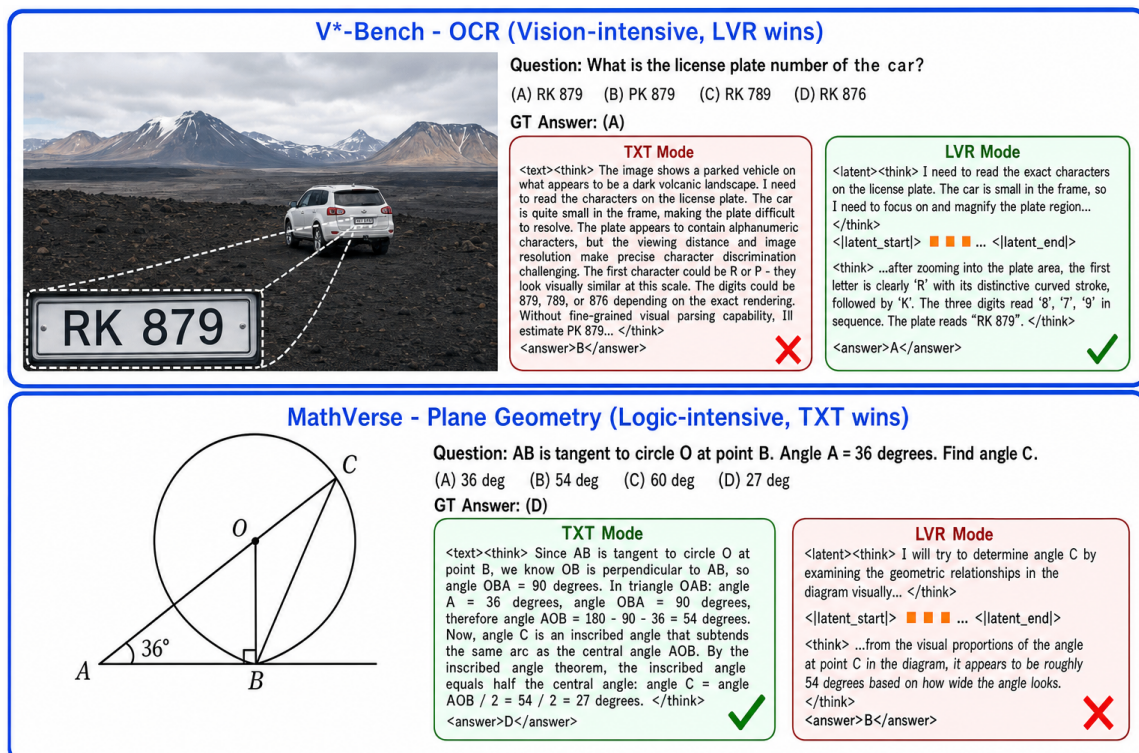


Figure 4: Case study comparing textual (TXT) and latent visual reasoning (LVR) modes on representative tasks. **Top:** A V*-Bench OCR task requiring fine-grained license plate recognition—textual reasoning fails to resolve characters at challenging viewing distance, while LVR zooms into the plate region in latent space and correctly identifies “RK 879”. **Bottom:** A MathVerse plane geometry problem requiring the inscribed angle theorem—LVR attempts visual estimation and produces an incorrect answer, while textual reasoning correctly applies geometric theorems to derive the answer.

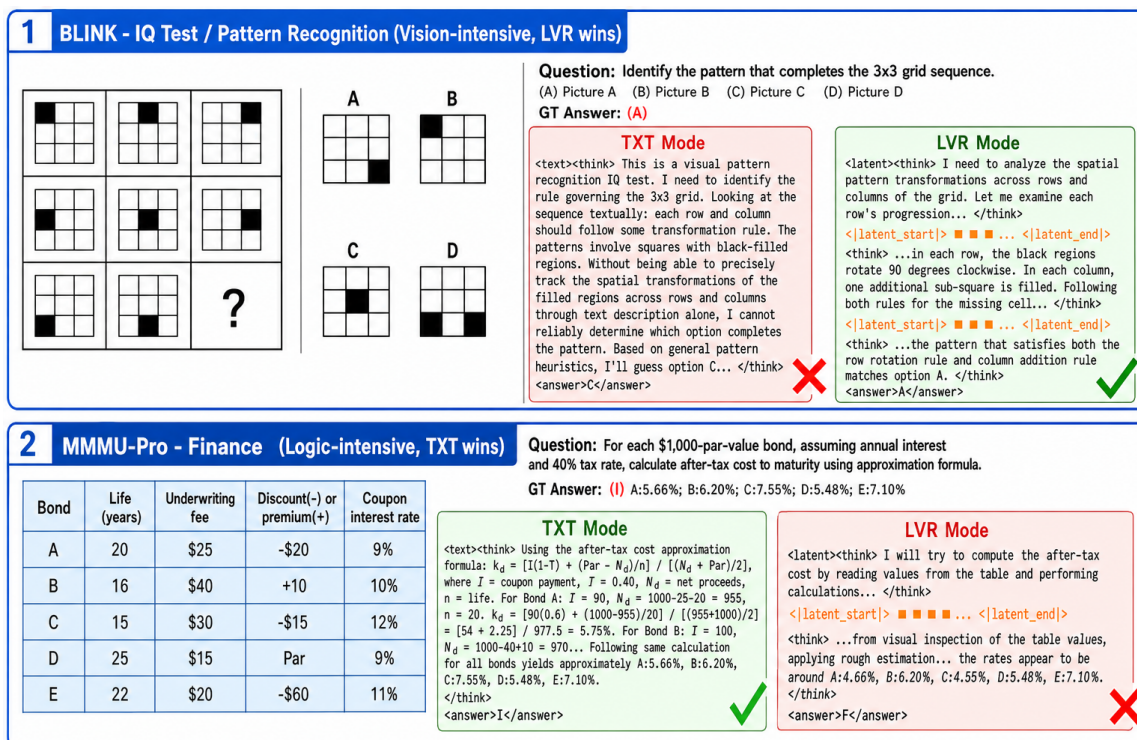


Figure 5: Additional case study examples. **Top:** A BLINK IQ test requiring visual pattern recognition across a 3×3 grid—textual reasoning cannot reliably track spatial transformations without visual computation, while LVR identifies the rotation and column addition rules to select the correct completion. **Bottom:** An MMMU-Pro finance problem requiring after-tax cost approximation—LVR attempts rough visual estimation from the table and fails, while textual reasoning correctly applies financial formulas to compute the answer.